

<b>Notice of Allowability</b>	<b>Application No.</b>	<b>Applicant(s)</b>	
	09/692,997	KLASK, KENNETH J.	
	Examiner Dennis G. Bonshock	Art Unit 2173	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS. This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1.  This communication is responsive to Applicant's Amendment filed on 2-2-2006 and the Proposed Amendment filed 4-12-2006.
2.  The allowed claim(s) is/are 22-26, 31-36, 41-44 and 49-54.
3.  Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
  - a)  All b)  Some\* c)  None of the:
    1.  Certified copies of the priority documents have been received.
    2.  Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
    3.  Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

\* Certified copies not received: \_\_\_\_\_.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.  
**THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.**

4.  A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.
5.  CORRECTED DRAWINGS ( as "replacement sheets") must be submitted.
  - (a)  including changes required by the Notice of Draftsperson's Patent Drawing Review ( PTO-948) attached
    - 1)  hereto or 2)  to Paper No./Mail Date \_\_\_\_\_.
  - (b)  including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date \_\_\_\_\_.

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
6.  DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

#### Attachment(s)

1.  Notice of References Cited (PTO-892)
2.  Notice of Draftperson's Patent Drawing Review (PTO-948)
3.  Information Disclosure Statements (PTO-1449 or PTO/SB/08),  
Paper No./Mail Date 2-2-06
4.  Examiner's Comment Regarding Requirement for Deposit of Biological Material
5.  Notice of Informal Patent Application (PTO-152)
6.  Interview Summary (PTO-413),  
Paper No./Mail Date \_\_\_\_\_.
7.  Examiner's Amendment/Comment
8.  Examiner's Statement of Reasons for Allowance
9.  Other \_\_\_\_\_.



**RAYMOND J. BAYERL**  
**PRIMARY EXAMINER**  
**ART UNIT 2173**

### **EXAMINER'S AMENDMENT**

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.
2. Authorization for this examiner's amendment was given in a telephone interview with David C. Hsia, on April 11, 2006.
3. The application has been amended as follows:

**- see attached claim amendment sheets submitted by Mr. Hsia**

*- Furthermore, applicant's representative as agreed to cancel previously withdrawn claims 45-48*

### **REASONS FOR ALLOWANCE**

1. The following is an examiner's statement of reasons for allowance:
2. The examiner considered the Applicant's Amendment filed on 2-2-2006 and the Proposed Amendment filed 4-12-2006, and after updated search, no other prior art of record has taught that which is presented in the amended claims (including the examiner amended claims).
3. Therefore, independent claims 22, 23, and 35, are found to be allowable along with the dependent claims 24-26, 31-34, 36, 41-44, and 49-54.
4. Independent claims 22, 23, and 35 when considered as a whole, are allowable over the prior art of record. Dye teaches, a system that uses a GUI

controller to provide for more efficient graphics management than a embedded controller alone (see column 2, lines 32-62), a screen that displays current information and allows for user input, the input can change a GUI property of the display, which is reflected on the display (see column 3, line 62 through column 4, line 20 and column 28, lines 19-34), the IMC generating video signals for driving the display, where the display (for which the user uses to interact with a user) is directly coupled to the IMC (see column 9, lines 59-67 and in figure 3C), the I/O (used to interact with the user) being coupled directly to the bus that the IMC is on (see column 10, lines 7-30 and in figures 2 and 3C), a system of using multiple buffers to provide the information for displaying the GUI on the screen and for buffering information that corresponds to a new interface (a change) (see column 21, lines 26-58), the use of opcode for attributes of objects or windows, whether it be for maintain a screen display or to effect a change to a different screen (see column 32, lines 33-55 and column 28, lines 13-25), a GUI library (see column 21, lines 27-36), sets of executable code (executable by the CPU) for rendering the display (see column 3, line 61 through column 4, line 7), graphics data stored in a frame buffer (see column 10, lines 59-61), a processor for handling inputs and rendering a GUI (see column 3, line 61 through column 4, line7), the processor connected to an input device and a digital to analog converter (see column 3, lines 27-43 and column 1, lines 37-37), the processor connected to memory via a memory bus (see column 2, lines 62-66), a CPU for executing codes for the objects and a frame buffer for rendering objects to the interface (see column 32, lines 33-55, column 11, lines 21-35, and column 10,

lines 59-61), and a pixel serialized coupled to the display for refreshing, and object rendering (see column 30, lines 34-55), and further teaches a graphics controller comprised in an integrated memory controller (IMC) which includes graphic processing capabilities to relieve workload from the main CPU (see column 3, lines 5-20, in column 2, lines 50-67 and column 1, lines 46-61). Dye teaches a graphics controller that is embedded in a memory controller, but doesn't specifically teach distinct system and graphics controllers each having their own controller specific memory. Margulis teaches a system using a graphics controller, similar to that of Dye, but further teaches, in column 2, line 62 through column 3, line 30 and in figure 2, a system with distinct system and graphics controllers each having their own controller specific memory. However, specifically the prior art of record fails to clearly teach or support the limitation of a graphics controller, separate from a system controller, that receives non-graphical data, via opcode, from the system controller and produces a GUI on the display using the non-graphical date, but generates the display based on all internal to the graphics controller graphics information. This graphics controller further accepting graphics information, as opcode, as input from an input device, rendering a second GUI object in response to input, and sending non-graphical data of the input to the system controller.

5. Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dennis G. Bonshock whose telephone number is (571) 272-4047. The examiner can normally be reached on Monday - Friday, 6:30 a.m. - 4:00 p.m.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cabeca can be reached on (571) 272-4048. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

4-13-06  
dgb



**RAYMOND J. BAYERL**  
**PRIMARY EXAMINER**  
**ART UNIT 2173**

**DRAFT**

Claim 22 (currently amended): A control system for a device, wherein a graphic user interface (“GUI”) controller operates a GUI of the device independently from an embedded controller of the device, the control system comprising:

the embedded controller for controlling and monitoring the device;

a liquid crystal display (“LCD”) for displaying the GUI to a user, the GUI including:

a first GUI object comprising a graphical presentation of a status of the device; and

a second GUI object comprising a graphical presentation of a command to the device;

a touch screen for detecting the command from the user;

the GUI controller, comprising:

at least one memory, comprising:

a document buffer storing a document defining an appearance of the GUI, the document comprising:

a first operation code (“opcode”) identifying the first GUI object;

a second opcode first operand of the first opcode, the first operand identifying a source of the status;

a third second opcode identifying the second GUI object; and

a fourth opcode second operand of the second opcode, the second operand identifying a destination of the command;

a data buffer storing the status and the command;

a GUI object library storing:

# DRAFT

a first set of executable codes defining an appearance and a functionality of the first GUI object, the first set of executable codes comprising instructions for rendering the first GUI object, receiving non-graphical data of the status from the embedded controller, and further rendering the first GUI object to show a visual response to a change to the status;

a second set of executable codes defining an appearance and a functionality of the second GUI object, the second set of executable codes comprising instructions for rendering the second GUI object, receiving the command from the touch screen, further rendering the second GUI object to show a visual response to a change to the command, and sending non-graphical data of the command to the embedded controller;

a frame buffer storing at least one complete display frame image of the GUI;

a GUI processor for rendering the GUI and handling user inputs independently from the embedded controller, wherein:

the GUI processor is coupled to the touch screen via a touch screen analog to digital converter to receive the command from the user, wherein the touch screen is not directly connected to the embedded controller;

the GUI processor is coupled to the embedded controller via a serial UART interface to send the non-graphical data of the command to the embedded controller and to receive the non-graphical data of the status from the embedded controller;

the GUI processor is coupled to the at least one memory via a memory bus interface, wherein:

in response to the first and the second opcodes opcode and the first operand, the GUI processor executes the first set of

# DRAFT

executable codes to render the first GUI object to the frame buffer independently from the embedded controller, to receive the non-graphical data of the status from the embedded controller, and to further render the first GUI object to the frame buffer to show a visual response to a change to the status independently from the embedded controller;

in response to the ~~third and the fourth opcodes second opcode and the second operand~~, the GUI processor executes the second set of executable codes to render the second GUI object to the frame buffer independently from the embedded controller, to receive the command from the touch screen independently from the embedded controller, to further render the second GUI object to the frame buffer to show a visual response to a change to the command independently from the embedded controller, and to send the non-graphical data of the command to the embedded controller;

a pixel serializer coupled to the LCD to continuously refresh the LCD with the complete display frame image in the frame buffer that contains both the rendered first GUI object and the rendered second GUI object.

**Claim 23 (currently amended): A first controller providing a graphical user interface ("GUT") for a device independently from a second controller that monitors and controls the device, the first controller comprising:**

at least one memory comprising:

a document buffer storing a document, the document comprising:

a first operation code ("opcode") identifying a first GUI object in the GUI ~~and a second opcode identifying a parameter of the GUI object, the first~~ GUI object being a graphical presentation of the a parameter, the parameter being from the second controller;

# DRAFT

a first operand of the first opcode, the first operand identifying the parameter of the first GUI object;

a second opcode identifying a second GUI object in the GUI, the second GUI object being a graphic presentation of a user input, the user input being from an input device; and

a second operand of the second opcode, the second operand identifying a destination of the user input;

a frame buffer for storing at least one complete frame of the GUI including the GUI object the first and the second GUI objects;

a GUI object library storing executable codes defining an appearance and a functionality of the GUI object the first and the second GUI objects, the executable codes comprising:

a first plurality of instructions for receiving a first non-graphical data of the parameter from the second controller and rendering the first GUI object in response to any change to the parameter;

a second plurality of instructions for receiving the user input from the input device, rendering the second GUI object in response to the user input, and sending a second non-graphical data of the user input to the second controller;

a processor coupled to the second controller and the at least one memory, wherein the processor (1) reads the first opcode, the first operand, the second opcode, and the second operand and the second opcodes, (2) reads the executable codes, (3) executes the executable codes to (a) receive the first non-graphical data of the parameter from the second controller, and to (b) render the first GUI object in response to any change to the parameter independently from the second controller, (c) receive the user input from the input device, (d) render the second GUI object in response to the user input independently from the second controller, and (e) send the second non-graphical data of the user input to the second controller, and (4) saves the rendered GUI object in the frame buffer.

# DRAFT

Claim 24 (previously presented): The first controller of claim 23, further comprising an output device coupled to the frame buffer to receive the GUI, the output device displaying the GUI to the user.

Claim 25 (previously presented): The first controller of claim 24, wherein the output device is a liquid crystal display ("LCD").

Claim 26 (previously presented): The first controller of claim 25, further comprising a pixel serializer coupled between the frame buffer and the LCD, the pixel serializer outputting each line of the GUI in the frame buffer to the LCD.

Claims 27 to 30 (canceled).

Claim 31 (previously presented): The first controller of claim 23, wherein the parameter is a status of the device from the second controller to the user.

Claim 32 (currently amended): The first controller of claim 31, wherein the first GUI object is a text field.

Claim 33 (currently amended): The first controller of claim 23, further comprising another memory coupled to the processor, the another memory storing the document, the processor loading the document from the another memory to the at least one memory.

Claim 34 (previously presented): The first controller of claim 23, wherein the second controller further comprises another memory storing the document, the second controller reading the document from the another memory and sending the document to the first controller, the first controller loading the document in the at least one memory.

Claim 35 (currently amended): A method for a first controller to generate a graphic user interface ("GUI") for a device independently from a second controller that monitors and controls the device, comprising:

loading a document into a document buffer in at least one memory, the document defining an appearance of the GUI, the document comprising:

a first operation code ("opcode") identifying a first GUI object and a second opcode identifying a parameter of the GUI object, the first GUI object being a

# DRAFT

graphical presentation of the a parameter, the parameter being from the second controller;

a first operand of the first opcode, the first operand identifying the parameter of the first GUI object;

a second opcode identifying a second GUI object, the second GUI object being a graphical presentation of a user input, the user input being from an input device;

a second operand of the second opcode, the second operand identifying a destination of the user input;

in response to the first opcode, the first operand, the second opcode, and the second operand and the second opcodes, retrieving executable codes of the GUI object the first and the second GUI objects from a GUI object library stored in the at least one memory, the executable codes defining an appearance and a functionality of the GUI object the first and the second GUI objects, the executable codes comprising:

a first plurality of instructions for receiving a first non-graphical data of the parameter from the second controller and rendering the first GUI object in response to any change to the parameter;

a second plurality of instructions for receiving the user input from the input device, rendering the graphic presentation of the second GUI object in response to the user input, and sending a second non-graphical data of the user input to the second controller;

independently from the second controller, executing the instructions executable codes to receive the first non-graphical data of the parameter from the second controller, and to render the first GUI object in response to any change to the parameter, receive the user input from the input device, render the second GUI object in response to the user input, and send the second non-graphical data of the user input to the second controller;

writing the rendered GUI object in a frame buffer in the at least one memory; and

**DRAFT**

sending the rendered GUI object from the frame buffer to an output device, wherein the output device displays the GUI to a user.

Claim 36 (previously presented): The method of claim 35, wherein the output device is a liquid crystal display (“LCD”).

Claim 37 to 40 (canceled).

Claim 41 (previously presented): The method of claim 35, wherein the parameter is a status of the device from the second controller to the user.

Claim 42 (currently amended): The method of claim 41, wherein said rendering the first GUI object comprises drawing a text field.

Claim 43 (previously presented): The method of claim 35, further comprising reading the document from another memory prior to said loading the document into the at least one memory.

Claim 44 (previously presented): The method of claim 35, further comprising receiving the document from the second controller prior to said loading the document into the at least one memory.

*(Canceled)*  
Claim 45 (*withdrawn*): A method for programming a graphic user interface (“GUI”) for a GUI controller of a device, comprising:

creating a HTML page defining an appearance of the GUI, said creating comprising:

writing an applet code for a GUI object;

writing a reference to a source of a parameter of the GUI object;

compiling the HTML page into a compact HTML page, said compiling comprising

replacing the applet code with a first operation code (“opcode”) identifying the GUI object;

replacing the reference with a second opcode identifying the parameter of the GUI object;

# DRAFT

reading the compact HTML page from at least one memory in the device; and  
in response to the first and the second opcodes, executing executable codes  
instructions in the at least one memory to communicate the parameter with the source  
and to render the GUI object in response to the parameter.

*Cancelled*  
**Claim 46 (withdrawn):** The method of claim 45, wherein said communicating the parameter with  
the source comprises receiving a status of the device from an embedded controller, the embedded  
controller monitoring and controlling the device.

*Cancelled*  
**Claim 47 (withdrawn):** The method of claim 45, wherein said communicating the parameter with  
the source comprises receiving a command from an input device.

*Cancelled*  
**Claim 48 (withdrawn):** The method of claim 47, further comprising executing the executable codes  
to send the command to the embedded controller.

**Claim 49 (new):** The first controller of claim 23, wherein the user input is a command from the user  
to the second controller for controlling the device.

**Claim 50 (new):** The first controller of claim 49, wherein the input device is one of a touch screen, a  
key pad, an infrared remote, and a voice decoder.

**Claim 51 (new):** The first controller of claim 50, wherein the second GUI object is one of a button  
and a list.

**Claim 52 (new):** The method of claim 35, wherein the user input is a command from the user to the  
second controller for controlling the device.

**Claim 53 (new):** The method of claim 52, wherein the input device is one of a touch screen, a key  
pad, an infrared remote, and a voice decoder.

**Claim 54 (new):** The method of claim 53, wherein said rendering the second GUI object comprises  
drawing one of a button and a list.